REMARKS

By this response, claims 21-41 are pending. Claims 21, 28, 32 and 37 are independent. Compared to prior versions, all claims remain as previously or originally presented. Claims 1-20 remain canceled. Substantively, the Examiner rejects claims 21-41 under 35 U.S.C. §101 as "lacking real world useful result[s]." She rejects claims 21-25, 26-29 and 31-41 as being obviously unpatentable over U.S. Patent No. 5,758,343 to Vigil in view of U.S. Patent Publication No. 2002/0032775 to Venkataramaiah et al. (hereafter, "VENKA," to use the Examiner's nomenclature). Claims 26 and 30 are rejected as obvious in view of the Vigil and VENKA combination, in further view of U.S. Patent Publication No. 2006/0129652 to Petrovskaya. The pending claims have many features and aspects not found in the combination of references, however, and reconsideration is respectfully requested.

Also, the Examiner either ignores claim limitations or rephrases them, incorrectly, to suit her needs. It cannot be stressed enough, though, that this is improper. In the MPEP §2143.03, the Examiner must find, in order to establish a *prima facie* case of obviousness, that all the claim limitations are taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). "**All words in a claim must be considered in judging the patentability of that claim against the prior art.**" *In re Wilson*, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970)(Emphasis added).

I. The present invention, generally:

The present invention relates broadly to computers, especially methods and apparatus for searching multiple disparate directories with a single, user-formulated query. Each of the claims require the notion that an administrator utility is configurable to associate directory classes of two or more disparate directories into a single user-searchable category so that users can later search the disparate directories with a single query.[1] The utility also preferably resides with a directory shell and users search from a directory browser. Searching convenience over the prior art is greatly enhanced.

The genesis for the instant invention stems from the unfulfilled need of a user being able to search multiple disparate directories with a single query of a user-defined category. For example, when two companies A & B join in a merger, each pre-merger company might include a human resources department with a directory listing of all the employees. When searching for information about an employee in one directory, a user would search the "Employee List" directory from company A and the "People Who Work Here" directory from company B. With the instant invention, the user can now search the merged company directory with a single query of a single category, such as "EMPLOYEES," because of the administrative pre-role that associates the two disparate directories ("Employee List" and "People Who Work Here") together in an administrator utility.

From the Applicant's specification, this example is typified as disparate or "different" directories that "potentially have different names for class attributes." *Appellant's Specification, p. 14, l. 27.* Representative "Cisco" and "Novell" corporate directories are given as disparate directories, including differing directory classes 51, that are searchable with a single query under an administrator-created user-searchable category 62 having the name "Find All." *Appellant's Specification, p. 8, l. 13 - p. 9, l. 4.* In turn, mapping or associating the differing directory classes 51 of the Cisco and Novell directories into a single user-searchable category 62 occurs, for example in a directory shell 60. In one embodiment, the "directory shell is queryable against the categories and category attributes to search and retrieve data of the objects in the directories." *Appellant's Specification, p. 5, ll. 20-22.* In another embodiment, the directory shell 60 "includes two aspects: an administrator utility and

[1] Of course, each of the independent claims is worded slightly differently than the other and the scope of any given claim is governed by its express language.

a directory browser." *Appellant's Specification, p. 11, ll. 14-15.* The administrator utility allows the administrator to disable or enable searching on a directory class by various mechanisms, such as checking a box (or not) under an Enabled Column of a table 122, for example. *E.g., Appellant's Specification, p. 12, l. 30 et seq.* User searching occurs, for example, via a query portion 210 of the directory browser shown as a page 200 in Figure 8. Results of the search are displayed in a variety of panels 220, 230 on the same page. Formats for both the utility and browser are preferably HTML.

II. Claims 21-41 are statutory subject matter under 35 U.S.C. §101. Among other things, they produce tangible and real-world useful results that particularly benefit the public and/or users.

The Examiner contends claims 21-41 violate the strictures of 35 U.S.C. §101 because they "lack [] real world useful results." *9-10-07 Office Action, p. 3, 2nd* ¶. Also, since they "do not have a readily well-established utility" or "particular benefit to the public or to the user(s)," the Applicant is "required" to distinguish the claims from the "three 35 U.S.C. 101 judicial exceptions (Laws of Nature, Natural Phenomena and Abstract Ideas)." *Id., citing MPEP 2106 IV C.*

First, it is inconsistent that the Examiner argues that the present invention is non-utilitarian, when the outstanding rejections are based, nearly entirely, on two United States issued patents. In other words, the Examiner attempts to explain how a certain portion of the invention is found in an issued patent to Vigil, while the remaining portion is found in an issued patent to VENKA. If true, though certainly not admitted, then the present invention should be utilitarian for the same exact reasons that Vigil and VENKA have both been found to be utilitarian. To argue otherwise is disingenuous.

Second, the law defines four categories of inventions, e.g., processes, machines, manufactures and compositions of matter. As mentioned under *MPEP §2106*, for example, the latter three categories define "things" or "products," while the first category, "processes," define "'actions' (i.e., inventions that consist of a series of steps or acts to be performed.)" Also, the U.S. Supreme Court has interpreted 35 U.S.C. §101 as broadly including "anything under the sun that is made by man." *Diamond v. Chakrabarty*, 447 US 303, 308-309, 206 USPQ 193, 197 (1980). Despite the seeming breadth of such an interpretation, it is more narrowly anything under the sun, made by man, other than the "laws of nature, natural phenomena, and abstract ideas." *Diamond v. Diehr*, 450 US 175, 185 (1981). The ultimate issue must also be dissected according to "wether the claim ***as a whole*** is drawn to statutory subject matter." *Emphasis added, AT&T Corp. v. Excel Communications Inc.*, 50 USPQ2d 1447, 1451 (Fed. Cir. 1999) (citing to *In re Alappat*, 33 F3d. 526, 31 USPQ2d 1545 (Fed. Cir. 1994), among other cites.).

In the present invention, however, nothing realistically can be construed as a law of nature, a natural phenomena or an abstract idea. Rather, the invention of the independent claims, as a whole, requires "a computer system" including a structural "directory shell" having both "an administrator utility" and "a directory browser." In turn, the utility contemplates "associat[ing] the directory class in the one of the directories to the directory class in the another of the directories, the result of associating the directory classes being a user-searchable category," while the browser provides a structure "whereby users can search the directory classes of the two or more disparate directories with a single query of the user-searchable category." Claim 32 even provides for "a table" in the utility configured to enable the associating of disparate directory classes and "a query portion" and "a panel" in the browser to actually conduct the search and view the results, respectively. In this manner, computer searching of dissimilar directories is made convenient, not only to users, but also to administrators responsible for associating the two or more disparate directories. The result is not only useful, but structurally tangible. Also, it is embodied in the many claims in apparatus and method format, including computer readable media with computer-executable instructions.

For at least the foregoing, the Applicant submits the claims are sufficient as written and in a utilitarian condition meeting 35 U.S.C. §101.

III. Even if Vigil and VENKA are properly combined, and the Applicant does not admit this, the two references do not result in the invention of claims 21-41.

A. Vigil, generally:

According to Vigil, methods and apparatus are taught for "providing an improved directory service which is highly scalable and which is able to handle large non-distributed directories." *Col. 3, ll. 7-9.* In Figure 13, for example, a "conventional directory information tree" (*col. 4, ll. 11-12*) includes entries corresponding to a root of the tree 10 and leaves. *Col. 1, ll. 28-40.* Each entry is also "unambiguously" identified according to its "distinguished name." *Col. 1, ll. 41-42.*

In Figures 1 and 2, however, various agents carve up the root and leaves in order to effectively canvass its entries from a search request of a client 109. (Purportedly, it overcomes the stated problem of the prior art whereby a single QUIPU agent can only handle about "40,000 entries" in a single directory (*col. 2, l. 28*) or about "100,000 entries in a single directory (*col. 2, l. 54*) per a QUIPU agent in combination with a "specialized delegate directory service agent." *Col. 2, ll. 37-38.*) Namely, a QUIPU directory service agent 100 (Figure 1) or 201 (Figure 2) "is responsible for storing and managing information at the top (near the root [200]) of the X.500 directory information tree" (*col. 4, ll. 43-45*), while "various delegate directory service agents [DSA] 101-104" maintain the "lower branches [e.g., leaves] of the directory information tree." *Col. 4, ll. 46-48.* Also, the QUIPU agent receives all requests and coordinates them amongst the many delegate agents by "chain[ing]." *Col. 4, l. 50.*

In other aspects, a directory interface shell (DISH) 106 "provides server-based access to the directory and provides a powerful interface onto the directory to give a user access to the full directory access protocol." *Col. 4, ll. 59-62.* An LDAP daemon 105 interfaces with the client 109 to "service directory requests originating from directory user agent (DUA) applications such as DUA application 108." *Col. 4, ll. 64-66.* In Figures 3-12, graphical user interface windows enable a "directory administrator" (*col. 6, l. 17*) to install various DSA's and coordinate their limitations in the tree (e.g., which leaves, leaf size, etc.) relative to the QUIPU, but without necessarily forcing an in-depth "understanding of the underlying configuration requirements of the QUIPU technologies used," (*col. 6, ll. 19-20*). *Figures 3-12, generally, Col. 6, l. 10 - col. 8, l. 9.*

At no time, however, does Vigil teach anything other than a single, homogenous directory, inclusive of roots, leaves, leaf entries, etc., that users query for search results. Its utility, to the extent Figures 3-12 show one, never enable the linking of disparate directories or linking them under a single user searchable category. Instead, it relates solely to carving-up the root and leaves of its single directory amongst multiple directory agents. Also, its directory shell (DISH 106) certainly never discloses an administrator utility as part thereof, much less one allowing the linking of disparate directory classes into a single category that users then search from a browser, that is also a part of the directory shell. As will be seen, it is these failures of Vigil that are unable to render the claims of the instant invention obvious.

B. VENKA, generally:

VENKA teaches, "[i]n contrast to traditional database models and their related persistence mechanisms which require that data be stored at a single physical location," the "effective[] transform[ation of] multiple directory servers into a single logical database." ¶ *[0023]*. As is seen, a Persistence Framework Service Manager (PFSM) 125 interfaces with a user 110 who makes a request Q, 115. In turn, the PFSM "dissects the persistence request 115 (into Q1 and Q2)" (¶ *[0029]*), which are individually and separately made (135, 150) of two isolated LDAP servers 140, 155 (e.g., one in Japan and one in the U.S. ¶ *[0038]*). Upon individual responses R1 and R2 being generated by the servers, they are returned 145, 160 to the PFSM. The PFSM "collects the pieces of data (R1 and R2) ... and returns a response 130 (R) to the user 110." ¶ *[0029]*. In this manner, "the persistence storage system and method allows the user of the network to effectively transform a plurality of directory servers into a single logical database." ¶*[0031]*.

At no time, however, does the PFSM ever make associations between the two LDAP servers for user searching, much less with an administrative utility associating categories of the servers into a single user-searchable category. In fact, it keeps the LDAP servers isolated and treats them quite separately. First, it "dissects" a comprehensive query Q into individual LDAP queries relative to the servers, e.g., Q1 and Q2, and then concatenates together individual responses (R1 + R2) before providing them back to the user. As will be seen, this is a failure of the reference, alone or combined with Vigil, relative to the obviousness rejections of the pending claims.

C. The combination of Vigil and VENKA never reveals a single user searchable category by way of an administrator utility that associates disparate directories together so that users can search the directory

classes of two or more disparate directories with a single query, much less from a browser of a directory shell, or a shell containing both the utility and the browser, as variously claimed in claims 21-41.

From above, neither Vigil nor VENKA teach any form of association between disparate directories. Vigil teaches nothing but a single, homogenous directory, carved up amongst (QUIPU and DSA) agents to effectively canvass the root and leaves of a tree of the single directory. VENKA keeps LDAP servers isolated by way of its separate query/response protocol with the PFSM. In turn, if neither reference individually teaches the limitations of the claims, the references together, in combination, also fail to teach them. For at least this reason, the claims are patentable as written and reconsideration is requested.

D. Even if properly combinable, Vigil and VENKA together only result in an ability to search isolated directories, with independent LDAP requests (VENKA) that become carved-up to canvass root and leaf concerns of each of the directories (Vigil). The result never meets the limitations of the claims.

Unequivocally, Vigil's teaching of multiple agents (QUIPU and DSA) to canvass the root and leaves of a directory, reside per a single server 107, Figure 1. To the extent VENKA then included Vigil's teaching, as suggested by the Examiner, in combination, VENKA's servers 140 and 155 would be each outfitted with the QUIPU and DSA agents to individually canvass roots and leaves per either server 140 or 155. This would not, however, reveal an association between the servers (140 and 155) themselves. Instead, the end result produces requests Q1 or Q2 from a PFSM, dissected from an original user request Q, that are then forward singularly to the directories of the servers 144, 155 as individual LDAP requests.

Q1 and Q2 compare, properly, to Vigil's request from its DUA application 108 to the LDAP Daemon 105, to search root/leaves with QUIPU agents 100 or 201 and delegate agents, e.g., DSA's 101-104, but nothing more.

For at least this reason, the claims are patentable as written and reconsideration is requested.

Even considering the possibility that Vigil's multiple, coordinated agent teaching (e.g., QUIPU and DSA's carving up roots and leaves in a single directory tree, respectively) could be moved beyond the boundaries of an individual server of VENKA (e.g., beyond servers 140 or 155), such as to VENKA's PFSM, (which the Applicant does not admit as being a proper combination), the two references still fail to render the present invention obvious. That is, Vigil only teaches a skilled artisan how to chain together items in a related, single directory, but cannot address concerns relative to disparate directories.

For example, Vigil's QUIPU agent is associated with a root of a tree, while the delegated agents (DSA's) are associated with the leaves. In turn, the root of Figures 2 and 13 corresponds to "c=<country>" and "o=<organization>," which is given as the US and GB (c=countries) and the University of Michigan, NASA and University College London (o=organization). The leaves, on the other hand, correspond to "ou," given as Faculty and Staff, Ames Research Centre, and Computer Science, and "cn," given as Timothy A Howes, Peter Yee and Paul Baker.

If there were a disparate directory, however, and none was given, countries may not be conveniently expressed in a hierarchical tree under the term "countries," or with a designator "c," but perhaps as "Member States," with a designator "MB." Similarly, the term organizations may not be conveniently given under the terms "organizations," or with the designator "o," but perhaps as "Entities," with a designator "e." Further, it may be that the disparate directory wouldn't even have a "country" hierarchical layer and its root would begin with the "Entities" layer in order to minimize the number of directory entries. (See, for example, the vastly different hierarchical layers in the Applicant's wire diagrams associated with directories 10A, 10b, and 10C, in its specification at Figure 1.) To this end, how do skilled artisans then rearrange the teaching of Vigil, in combination with VENKA, to arrive at the present invention? The answer is they cannot and do not, since Vigil gives absolutely no insight as to how to accomplish an association amongst disparate directories.

It seems the Examiner is engaged in hindsight analysis, clearly improper under the law. Namely, the Examiner finds a directory shell and an administrator utility in Vigil and multiple servers in VENKA, that are searched from a request of a user, and then concludes the precise limitations of the claims are obvious in view of the two references. However, she contorts some limitations and ignores others (as will be seen below). As is known, the "[d]etermination of obviousness cannot be based on the hindsight combination of components selectively culled from the prior art to fit the parameters of the invention." *ATD Corp. v. Lydall, Inc.*, 159 F.3d 534, 536, 48 USPQ2d 1321 (Fed. Cir. 1998). Also, rejections of this sort are clearly discouraged.[2]

Individually, each of the pending claims define themselves over both references for one or more of the reasons given hereafter.

Claim 21: This claim requires a directory shell able to reference two or more disparate directories. The shell includes both an administrator utility and directory browser. The utility is configurable to "associate" the directory classes of the two or more disparate

[2] As is well established, "virtually all [inventions] are combinations of old elements." *Ruiz v. A.B. Chance Co.*, 69 USPQ2d 1686, 1690 (Fed. Cir. 2004). Also, an obvious determination under 35 U.S.C. 103(a) requires an "as a whole" analysis of the prior art to otherwise prevent an impermissible "evaluation of the invention part by part." *Id.* For otherwise, "an obviousness assessment might break an invention into its component parts (A+B+C), then find a prior art reference containing A, another containing B, and another containing C, and on that basis alone declare the invention obvious." *Id.* In turn, "this form of hindsight reasoning, using the invention as a roadmap to find its prior components, would discount the value of combining various existing features or principles in a new way to achieve a new result - often the very definition of invention." *Id.*

directories into "a user-searchable category." The browser is then a structure where users search the directory classes of the two or more disparate directories with "a single query" of the user-searchable category.

Vigil and VENKA, alone or in combination, do not render this obvious. **Vigil teaches nothing but a single, homogenous directory (not multiple disparate directories)**, carved up amongst (QUIPU and DSA) agents to effectively canvass the root and leaves of a tree of the single directory. VENKA keeps LDAP servers isolated by way of its separate query/response protocol with the PFSM. To the extent VENKA then included Vigil's teaching, or vice versa, VENKA's servers 140 and 155 would be each outfitted with QUIPU and DSA agents to individually canvass roots and leaves per either server 140 or 155. Nowhere does this meet the limitations of the claim.

Also, the Examiner severely mis-characterizes Vigil! It is contended Vigil teaches "a directory shell able to reference two or more disparate directory." *9-10-07 Office Action, p. 5, 2nd ¶*. However, this is not true. Only a single, homogeneous directory is ever referenced by Vigil. **The fact that different roots and leaves are mentioned does not translate into disparate directories, but a hierarchical tree in a single directory**. She further states the claimed "administrator utility with the directory shell" that is "configurable to associate the directory class in the one of the [disparate] directories to the directory class in the another of the [disparate] directories" is found in Vigil by Figures 3-12 showing "setup and manag[ement of] the multiple directory server agents." *Id.* However, setting-up DSA's in Vigil nowhere translates to the claimed limitation as shown above.

Claims 22-27: These claims depend from claim 21 and further require nuances of the system. Namely, they specify the whereabouts of the two or more disparate directories and the directory shell (claim 22); they describe structure that sends the single query and allows communication between the structure and the disparate directories (claims 23 and 24); they describe the category as including a category attribute mapped to class attributes of the directory class (claim 25); and they define where search results are displayed in the browser and where associating occurs in the utility (claims 26 and 27). Vigil and VENKA, however, fail as references in this regard, especially considering the limitations of parent claim 21.

The Examiner's rejection particularly fails since Vigil does not teach, as contended in claim 22, "wherein the two or more disparate directories are managed on a plurality of servers in communication with a computer onto which the directory shell is loaded." *9-10-07 Office Action, p. 6, penultimate* ¶. Unequivocally, the server 107 of Vigil's arrangement is that where the directory shell is loaded, and no other servers are taught in the management of the directory tree.

As contended by the Examiner in claim 24, Vigil supposedly teaches "a director driver for each of the two or more disparate directories to allow the directory interface to communicated therewith (figs. 1 and 2, col. 6, lines 10-67)." *9-10-07 Office Action, p. 7, final* ¶. **Nothing in all of Vigil's column 6, however, mentions a "driver."** Column 6 mostly teaches GUI's.

Claim 28: This claim describes methodology of "providing" the directory shell (the shell including both the utility for administrators and the browser from where users search) to "enable" the administrator utility to "associate" directory classes of two or more disparate directories into a single user-searchable category. Searching of the category then occurs from the directory browser of the shell. Searching is also done in a "direct" manner with a "single" query.

Vigil, however, teaches nothing but a single, homogenous directory, carved up amongst (QUIPU and DSA) agents to effectively canvass the root and leaves of a tree of the single directory. VENKA keeps LDAP servers isolated by way of its separate query/response protocol with the PFSM. To the extent VENKA then included Vigil's teaching, or vice versa, VENKA's servers 140 and 155 would be each outfitted with QUIPU and DSA agents to individually canvass roots and leaves per either server 140 or 155.

Also, to the extent Vigil teaches a single query, only a single directory gets searched. To the extent VENKA teaches a single query, VENKA takes a first query Q of a user and then "dissects" it into two components Q1 and Q2 for each of servers 140 and 155. This hardly, then, equates to "direct" searching of "two or more disparate directories," much less direct searching by way of the user-searchable category created by associating the disparate directories. **The Examiner completely ignores this "direct searching" limitation and, if for no other reason, her *prima facie* rejection fails.**

Claims 29 and 30: These claims relate to specifying how the utility of claim 28 is enabled to associate directory classes and to displaying search results in a panel of the browser, in turn, in the directory shell containing both the utility and browser. Neither Vigil nor VENKA intimate such features, especially considering these claims further include the features of claim 28.

Also, the Examiner states relative to claim 29 that "Vigil does not explicitly teach mapping a category attribute of the single user-searchable category to one or more class attributes of the directory class," and looks to VENKA to supply the missing teaching. *9-10-07 Office Action, p. 9, last line - p. 10, first two lines*. But see, the Examiner made the following rejection of claim 25: "Vigil teaches wherein the user-searchable category includes a category attribute mapped to one or more class attributes of the directory class." *9-10-07 Office Action, p. 8, 1st* ¶. **On their face, these two positions of the Examiner cannot be harmonized. The written record fails.**

Claim 31 relates to computer readable media having executable instructions for performing the methodology of claim 28.

Claim 32: This claim is similar to claim 21 and is patentable for the reasons given above. However, it further builds on claim 21and requires a table, a query portion and a panel in the utility and browser for associating, searching and displaying search results. Vigil and VENKA are far short of rendering this obvious. Neither reference teaches a shell having both an administrator utility for associating the disparate directories and a browser from where users search the associated directories, much less a table with the utility, or a query portion and a panel in the browser for searching and displaying search results. Searching of the category is also done in a "direct" manner with a "single" query.

To the extent Vigil teaches a single query, only a single directory gets searched. To the extent VENKA teaches a single query, VENKA takes a first query Q of a user and then "dissects" it into two components Q1 and Q2 for each of servers 140 and 155. This hardly, then, equates to "direct" searching of "two or more disparate directories," much less direct searching by way of the user-searchable category created by associating the disparate directories. **Again the Examiner completely ignores this "direct searching" limitation and, if for no other reason, her *prima facie* rejection fails.**

Furthermore, the Examiner rejects claim 30 in further view of Petrovskaya since "Vigil and VENKA do not teach users can view search results." *9-10-07 Office Action, p. 15, 5th* ¶. But, see, claim 32 which precisely requires "a panel in the directory browser where users can view search results of the single query," and such is rejected as purportedly found in the Vigil and VENKA combination. **Again, the Examiner fails the written record. These two positions of claim 30 and 32 cannot be harmonized.**

Claims 33-36: More narrowly, these claims build on claim 32 and particularly specify panels for the browser, check boxes for associating, an enable column in the table and an HTML format for both the utility and browser.

Claim 37: This claim requires the "creation" of a single user-searchable category from directory classes of two or more disparate directories and an ability to "directly" search these classes with a "single query" of the category. Negatively, the claim further requires the absence or any creation or use of a virtual "or other" directory. Neither Vigil nor VENKA teach this single searchable category created from disparate directories. Vigil teaches single directories, carved up amongst agents into a pseudo root portion of the directory and a leave portion, e.g., a "virtual" or "other" directory. VENKA teaches multiple queries of LDAP servers dissected from a user request.

Also, to the extent Vigil teaches a single query, only a single directory gets searched. To the extent VENKA teaches a single query, VENKA takes a first query Q of a user and then "dissects" it into two components Q1 and Q2 for each of servers 140 and 155. This hardly, then, equates to "directly" searching "two or more disparate directories," much less directly searching by way of the user-searchable category created by associating the disparate directories. **Once again, the Examiner completely ignores this "directly searching" limitation and, if for no other reason, her *prima facie* rejection fails.**

Claims 38-40: These claims all relate to how a category of claim 37 is created. In one aspect, "the creating" further includes associating directory classes in the utility (claim 38). In another, it relates to creating more than one category (claim 39). In still another, it requires providing a directory shell for loading on a computer in communication with servers having the disparate directories (claim 40). **For claim 40, Vigil's shell (DISH) loads on the server 107 having the single directory, not on a computer in connection with the server. VENKA has no shell.**

Claim 41 relates to computer readable media having executable instructions for performing the methodology of claim 37.

The dependent claims are also patentable for their reliance on one or more of the independent claims 21, 28, 32 or 37. Thus, no direct discussion of Petrovskaya is necessary at this time, with the *proviso* that the Applicant reserves its rights to make later arguments regarding its failure as a teaching, its improper combination with either Vigil, VENKA, or both, or for additional reasons. As the law provides, if an independent claim is non-obvious under 35 U.S.C. 103, then any claim depending therefrom is non-obvious. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).

IV. Conclusion

The Appellant submits that: (1) all claims are in a condition for allowance; (2) all claims are statutory under 35 U.S.C. §101; and (3) claims 21-41 are non-obvious variants under 35 U.S.C. §103(a) regarding Vigil, VENKA and/or Petrovskaya. Accordingly, it is respectfully requested that the rejections of the pending claims be reversed and the application allowed.

To the extent any fees are due, although none are believed due, the undersigned authorizes their deduction from Deposit Account No. 11-0978.

Respectfully submitted,

KING & SCHICKLI, PLLC

Michael T. Sanderson
Reg. No. 43,082

247 North Broadway
Lexington, KY 40507
(859) 252-0889